

# Fast Depth Level Range Determination Algorithm for 3D-HEVC System Application

Hanqing Ding<sup>a</sup>, Wei An<sup>a</sup>, Tao Yan<sup>b</sup>, and Qiuwen Zhang<sup>a,\*</sup>

<sup>a</sup>College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, China

<sup>b</sup>School of Information Science and Technology, Jiujiang University, Jiujiang, 332005, China

---

## Abstract

To alleviate 3D-HEVC computational burden, a novel depth level range determination method based on motion homogeneity is proposed in this paper. Since the multi-view videos show the same frame at the same time, it is important to use the selected depth levels and prediction modes to save coding time since they are highly multi-view content-dependent. The motion homogeneity model for CU is characterized by motion vectors and prediction modes with corresponding CUs in the reference views. According to this model, we propose a fast depth level range determination method. The experimental results show that the fast method is able to reduce the computation time by an average of 30.9%, while maintaining a similar rate-distortion (RD) value compared to the initial 3D-HEVC system.

**Keywords:** 3D-HEVC system; motion characteristics; depth level

(Submitted on May 16, 2018; Revised on June 15, 2018; Accepted on July 16, 2018)

© 2018 Totem Publisher, Inc. All rights reserved.

---

## 1. Introduction

The recent high efficiency video coding (HEVC) system attains a better rate-distortion (RD) efficiency compared to H.264 (50% bitrate savings for equivalent perceptual video quality) by utilizing new inter and intra prediction modes [1-2]. Following this, the ITU-T VCEG and ISO/IEC MPEG standardization bodies published the joint collaborative team on 3D video coding extension development (JCT-3V) to develop the next-generation 3D video coding-based HEVC (3D-HEVC) system [3]. 3D-HEVC utilizes a modified HEVC encoder to encode texture video and its connected depth map, introducing new techniques based on coded frames [4]. Disparity-compensated prediction, inter-view motion prediction, advanced residual prediction, illumination compensation, and view synthesis prediction (VSP) are presented in [5]. Therefore, a computationally expensive mode prediction must check whole candidate prediction modes and depth levels to select the most suitable mode by using the Lagrange multiplier. Those additional coding tools have a high texture video coding efficiency, but they also result in troubles that obstruct applications of 3D-HEVC. Based on the above analysis, it is important to develop a fast texture video coding method and alleviate the computational burden of the 3D-HEVC system with minimal loss of 3D video quality.

## 2. Related Work

A content-aware prediction algorithm [6] achieves the computation complexity saving of MVC and features views correlation among multiple views. Since the video contents in different views are highly related, unnecessary coding processes can be omitted based on inter-view mode decisions. An inter mode prediction to reduce encoding time by utilizing textural segmentation and correlations is reported in [7]. However, the characteristics of the inter mode selection method in 3D-HEVC are different from those of MVC. An early DIRECT mode model [8] for MVC mode decision can reduce computational burden between 79.57% to 89.21% with little RD reduction. A fast mode decision algorithm utilizing stopping theory is investigated in [9] with a balance between time and performance in MVC. However, these mode

---

\* Corresponding author.

E-mail address: zhangqwen@126.com

reduction methods cannot be used for the 3D-HEVC mode decision because the number of prediction modes is increased. Moreover, the key factors in these methods of MVC are not considered and cannot work well when encountering the increased number of PU prediction in 3D-HEVC. Therefore, 3D-HEVC can save computational complexity efficiently by taking the advantage of the new mode prediction.

To this end, a few approaches are developed to alleviate the computational burden of texture video. A low complexity residual prediction method has been designed in [10] to ensure that 3D-HEVC is suitable for hardware, and it aims at reducing memory access and computations. An efficient online-learning-based scheme is employed in [11] for the dependent texture views. In addition, this method can adaptively adjust the motion search range and then simplify the inter-intra prediction mode search. However, this method should search for a threshold by training texture video data online, and a significant challenge is the accuracy of the untrained texture video. Fast coding algorithms are developed in [12] and utilize relevant information from depth map-texture video and inter-view correlations. These algorithms only utilize correlation of the texture video-depth map, inter-view, and spatial-temporal.

To improve these limitations of 3D-HEVC, we propose a fast texture video based on motion homogeneity classification. Because the multi-view texture videos' motion characteristics are very content-dependent, it is useless for employing an immobile motion search range and depth levels range for the entire encoding process. Therefore, it is desired to omit some unnecessary depth levels in the 3D-HEVC mode decision.

### 3. Proposed Fast Depth Level Range Determination Algorithm

3D-HEVC has a fixed CU size, and the largest size is up to 64. The corresponding level range is 3. All CU depth levels must always fully search in 3D-HEVC. However, the depth level of "0" is always applied to static or homogeneous region coding and hardly chosen for treeblocks, which have complex motion. Thus, the depth level range can be adaptively skipped utilizing the advantage of motion homogeneity property in texture video blocks.

Based on these observations, it is necessary to analyze the motion vector of the current treeblock using the motion information in the spatial-temporal and the previous coded view corresponding treeblocks. A motion predictor ( $\Omega$ ) is defined in function (1) and contains three types of motion predictors:

$$\Omega = \{M_s, M_T, M_v\} \quad (1)$$

Where  $M_s$  denotes spatial motion predictors in the current texture view (including  $M_1$ ,  $M_2$ ,  $M_3$ , and  $M_4$  in Figure 1);  $M_T$  denotes temporal motion predictors in the previously coded frame (including  $M_5$ ,  $M_6$ ,  $M_7$ ,  $M_8$ , and  $M_9$ ) lying on the same position as the current texture treeblock  $M_s$ ; and  $M_v$  denotes the inter-view motion predictor in the previously coded view (including  $M_{10}$ ,  $M_{11}$ ,  $M_{12}$ ,  $M_{13}$ , and  $M_{14}$ ) located at the same position as the  $M_s$  in Figure 1.

To verify motion information correlation between the spatial-temporal and the coded view corresponding treeblocks, an extensive experiment is performed by employing recommended test video sequences with different resolution and motion information. Experimental environment: I-B-P view structure, QPs with 25, 30, 35, and 40; group of pictures (GOP) size = 8; treeblock size = 64; depth level range up to 3; search range of motion estimation (ME) is 64, context-adaptive binary arithmetic coding (CABAC) entropy coding; 100 test frames for each sequence. Tables 1-3 prove the correlations of the current treeblock in spatial-temporal and inter-view neighboring treeblocks. The motion data (Motion Vector (MV)) connection among the current and neighboring treeblocks in spatial-temporal and inter-view is defined in function (2):

$$Corr = \frac{\sum_{i=0}^{i=D} (Treeblock_{cur\_i} - Treeblock_{inter\_i} (Treeblock_{spa\_i} / Treeblock_{tem\_i}))}{D} \quad (2)$$

Where  $Corr$  is set to 0 to 1, which shows the coding information correlation degree.  $D$  and  $i$  are the max and current depth level of the current treeblock, respectively;  $Treeblock_{cur\_i}$  and  $Treeblock_{tem\_i}$  are the current and temporal adjacent treeblock at the  $i^{th}$  level;  $Treeblock_{tem\_i}$  and  $Treeblock_{spa\_i}$  are the temporal adjacent at same view and spatial

treeblock of the current treeblock at identical time, respectively; and  $Treeblock_{inter\_i}$  is the co-located treeblock in the previously coded base view frames. The previously coded view adjacent treeblock  $Treeblock_{inter\_i}$  (in predictors  $M_V$ ), the spatially adjacent treeblock  $Treeblock_{spa\_i}$  (in predictors  $M_S$ ), and temporally adjacent treeblock  $Treeblock_{tem\_i}$  (in predictors  $M_T$ ) are described in Figure 1. Equation (2) is performed on the pixel level. Experimental results demonstrate that it is reasonable for the coding information of the treeblocks with current, spatial-temporal, and coded view to have high correlation.

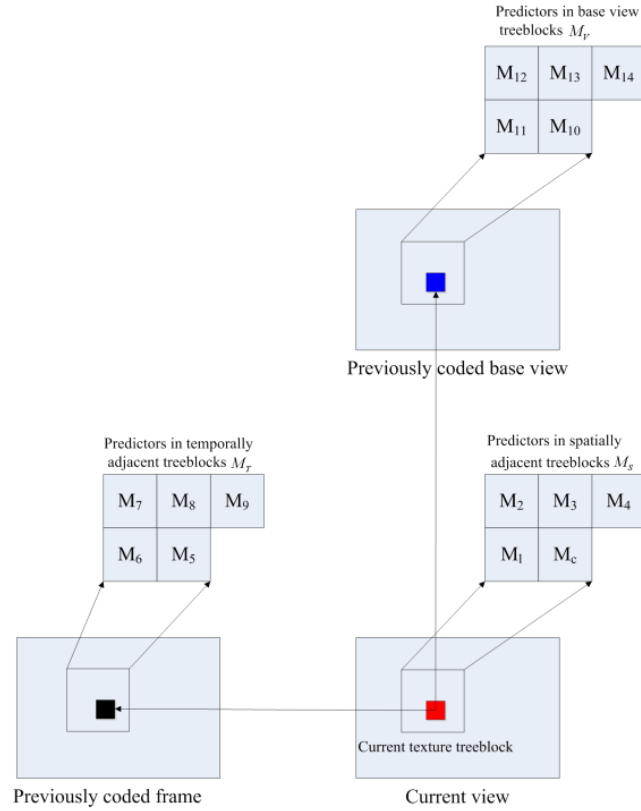


Figure 1. Predictors of the current treeblock and neighboring treeblocks

Table 1. Motion information correlation of the current texture treeblock and its spatial neighboring treeblocks

Index in Figure 1	$M_1$	$M_2$	$M_3$	$M_4$
Correlation	0.783	0.731	0.801	0.713

Table 2. Motion information correlation of the current texture treeblock and its temporal neighboring treeblocks

Index in Figure 1	$M_5$	$M_6$	$M_7$	$M_8$	$M_9$
Correlation	0.623	0.604	0.572	0.592	0.557

Table 3. Motion information correlation of the current texture treeblock and its inter-view neighboring treeblocks

Index in Figure 1	$M_{10}$	$M_{11}$	$M_{12}$	$M_{13}$	$M_{14}$
Correlation	0.712	0.692	0.668	0.682	0.649

Based on the motion predictors  $\Omega$  in Equation (1), the motion information from the spatial-temporal and inter-view adjacent predictors is extracted to analyze ME characteristics of the current texture treeblock in 3D-HEVC. Thus, an efficient criterion is needed to identify the motion homogeneity degree of the current texture treeblock between the adjacent treeblocks. It is explained as follows: motion vectors from the current texture treeblock and those from the  $4 \times 4$  block covered by associated treeblocks (all spatial-temporal and inter-view adjacent treeblocks:  $M_1, M_2, \dots, M_{14}$  in Figure 1) are exploited to calculate the motion information. Thus, a motion vector with  $4 \times 4$  block level is performed to calculate the motion homogeneity of texture video. We hypothesize that a texture treeblock lying on the  $r$ th row and  $w$ th column is

expressed as  $TB_{rw}$ , and the motion vectors of  $4 \times 4$  treeblocks are defined as  $MV_{ij} = (MVx_{ij}, MVy_{ij})$ ,  $i \in [4r, 4r+3]$ , and  $j \in [4w, 4w+3]$ . The average error of the forward motion vectors at  $4 \times 4$  block level covered by the adjacent treeblocks listed in Figure 1 can represent the motion homogeneity of the current texture treeblock. The motion homogeneities of the texture treeblock in the  $x$  and  $y$  directions are described separately.

$$MHx = \frac{1}{N} \sum_{(i,j) \in \Omega} \left| MVx_{ij} - \alpha_{ij} \cdot \sum_{(i,j) \in \Omega} MVx_{ij} \right| \quad (3)$$

$$MHy = \frac{1}{N} \sum_{(i,j) \in \Omega} \left| MVy_{ij} - \alpha_{ij} \cdot \sum_{(i,j) \in \Omega} MVy_{ij} \right| \quad (4)$$

Where  $MHx$  and  $MHy$  are the motion with horizontal  $x$  and vertical  $y$  components,  $N$  denotes the sum values of the  $4 \times 4$  block covered by the current and its neighboring treeblocks (all spatial-temporal and inter-view adjacent treeblocks  $M_s, M_t, M_v : M_1, M_2, \dots, M_{14}$  as in Figure 1). For our experiments, we have used  $N = (14+1) \times 4 \times 4 = 240$ .  $\alpha_{ij}$  is the motion-weight factor, and it is allotted to the adjacent treeblocks in spatial-temporal space based on the correlation. The stronger the motion correlation, the larger the weight. Meanwhile, the motion-weight factors have a specific property,  $\sum \alpha_{ij} = 1$ .  $\alpha_{ij}$  is defined based on the effect of adjacent treeblocks on the current treeblock. According to the motion information correlation (the current treeblocks and  $M_s, M_t, M_v$ ) in Tables 1-3, the factor  $\alpha_{ij}$  is shown in Table 4. These values ( $\alpha_{ij}$ ) are set based on a wide range of experiments, and these values achieve good coding results on test 3D sequences with different motion features. Thus, the motion homogeneity classification parameter is calculated in function (5):

$$MH = MHx + MHy \quad (5)$$

Table 4. The motion-weight factors assigned to each neighboring treeblock in predictors  $\Omega$

Index in Figure 1	$M_1$	$M_2$	$M_3$	$M_4$	$M_5$	$M_6$	$M_7$
$\alpha_{ij}$	0.10	0.10	0.10	0.10	0.05	0.05	0.05
Index in Figure 1	$M_8$	$M_9$	$M_{10}$	$M_{11}$	$M_{12}$	$M_{13}$	$M_{14}$
$\alpha_{ij}$	0.05	0.05	0.07	0.07	0.07	0.07	0.07

According to the motion homogeneity parameter  $MH$ , a treeblock can be classified into two kinds, static homogenous treeblocks and motion complex treeblocks. Set  $T_c$  as the current texture treeblock, and the decision is defined as follows:

$$T_c \subset \begin{cases} \text{Static homogenous region,} & \text{if } MH < Thr \\ \text{Motion complex region,} & \text{otherwise} \end{cases} \quad (6)$$

Where  $Thr$  are threshold factors.  $Thr$  can be used to decide whether a treeblock belongs to a static homogenous or a motion complex region. If the parameter  $MH$  is smaller than  $Thr$ , the current texture treeblock is considered to be in a region of static homogenous; otherwise, the current texture treeblock is in a motion complex region. In fact, the real world 3D video sequence is characterized by multiple edges and large areas of the background. Therefore, the probability of static and slow motion is higher compared to complex motion in real-world 3D videos. We focus separately on static and slow motion to save as much encoding time as possible. Based on the above discussion, we use the average motion homogeneity of each treeblock to determine the threshold  $Thr$ . The average motion homogeneity of each treeblock  $MH_{average}$  is calculated as follows:

$$MH_{average} = \frac{1}{R \times C} \cdot \sum_{r=0}^{R-1} \sum_{c=0}^{C-1} (MHx_{r,c} + MHy_{r,c}) \quad (7)$$

Where  $R$  and  $C$  represent the sum of the treeblocks in a row and in a column correspondingly. According to experiments, the  $Thr$  is  $0.7 \cdot MH_{\text{average}}$ , which achieves a good effect on different sequences and is stable for the QP level in the 3D-HEVC system.

With the motion homogeneity of the current treeblock (based on Equation (6)), each treeblock can be classified into two types, static homogenous treeblocks and motion complex treeblocks. Table 5 shows the depth level proportions for two kinds of texture video treeblocks in the exhaustive mode decision process.

Table 5. Texture video depth level distribution for two types treeblocks

Sequences	Treeblocks in static homogenous region				Treeblocks in motion complex region			
	Level0 (%)	Level1 (%)	Level2 (%)	Level3 (%)	Level0 (%)	Level1 (%)	Level2 (%)	Level3 (%)
Kendo	65.2	31.7	2.1	1.0	1.9	27.2	31.7	39.2
Balloons	67.1	29.5	2.6	0.8	1.4	28.7	32.3	37.6
Newspaper	59.4	35.7	3.2	1.7	2.2	33.9	35.7	28.2
Shark	56.7	40.6	1.8	0.9	0.6	22.7	28.6	48.1
Undo_Dancer	57.8	38.6	2.5	1.1	0.5	22.4	30.2	46.9
GT_Fly	64.1	33.2	1.9	0.8	1.5	29.2	32.5	36.8
Poznan_Street	66.2	31.3	1.9	0.6	1.7	25.6	34.5	38.2
Poznan_Hall2	72.5	26.1	1.2	0.2	3.3	38.4	34.2	24.1
Average	63.6	33.3	2.2	0.9	1.6	28.5	32.5	37.4

“Level 0”, “Level 1”, “Level 2”, and “Level 3” represent depth levels of the texture video treeblock. For treeblocks in a static homogenous region, about 63.6% encoded as level “0”, while 33.3% encoded as “1”. Therefore, when the maximum level is “1”, 96.9% of texture video treeblocks are covered. Then, the mode prediction levels of “2” and “3” ( $16 \times 16$ ,  $8 \times 8$ ) will be omitted. For treeblocks with complex motion, about 98.4% encoded as “1”, “2”, and “3” ( $32 \times 32$ ,  $16 \times 16$  and  $8 \times 8$ ). If the depth level is between “1” and “3”, almost 98.4% of texture video treeblocks are covered. Therefore, only 1.6% of treeblocks select “0”, and the level of “0” ( $64 \times 64$ ) will be omitted. With the aforementioned features, the candidate depth levels are summarized in Table 6. According to the depth level range determination algorithm, most texture video treeblocks will skip some unnecessary levels. The experimental results show that the candidate depth levels are limited, and the computational complexity of 3D-HEVC is reduced significantly.

Table 6. Candidate depth levels of texture video coding for two types of treeblocks

Texture video treeblock type	Candidate depth levels	Depth range [ $Depth_{\min}$ , $Depth_{\max}$ ]
Static homogenous region	0, 1	[0,1]
Motion complex region	1, 2, 3	[1,2,3]

#### 4. Experimental Results

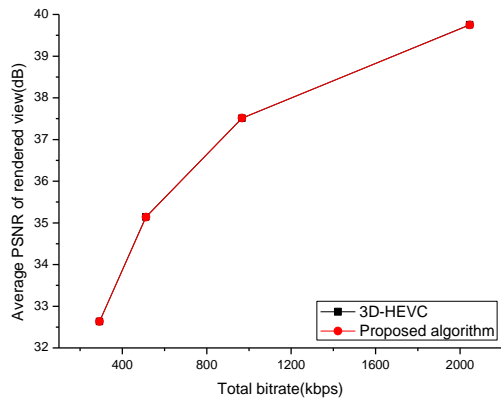
For evaluating the rationality of the above analysis, we perform indispensable experiments on the 3D-HEVC test model (HTM 16.0). We have tested eight sequences in two resolutions ( $1024 \times 768$ ,  $1920 \times 1088$ ) recommended by the JCT-3V Group under common test conditions (CTC) configuration.

Table 7 shows that our proposed algorithms could cut back the coding time without hampering potency. 30.9% encoding time reduction is attained by employing our algorithm. The decrease in coding time is vast for the sequences with large global motion of “Shark” (37.2%) and “Undo\_Dancer” (36.5%), but it is also evident for the small motion variation sequence of “Poznan\_Hall2” (23.7%). Meantime, this method demonstrates little efficiency loss, and the bitrate increase is 0.72 % (0.02 dB PSNR drop) for texture videos and 0.29% (0.01 dB PSNR drop) for rendered views. Accordingly, our algorithm will omit meaningless depth levels in the 3D-HEVC system.

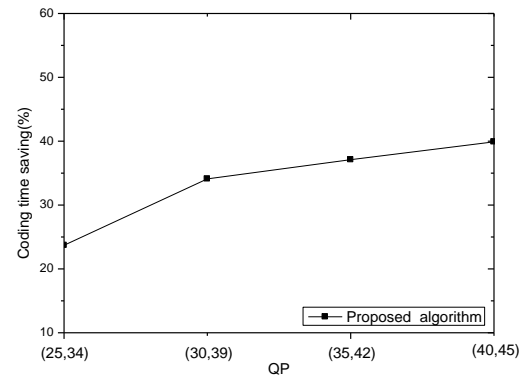
Figure 2 shows RD and time-saving curves compared to 3D-HEVC for two typical sequences, “Newspaper” and “Shark”, in “Rendered view”. Our algorithm reduces coding time over a large bitrate range without hampering PSNR or enhancement in bitrate. Furthermore, the encoder runtime will save more effectively as the bitrate decreases. It is reasonable that the probability of testing “0” and “1” using the proposed algorithm increases as the QP increases.

Table 7. Coding results of the proposed algorithm compared to the original 3D-HEVC

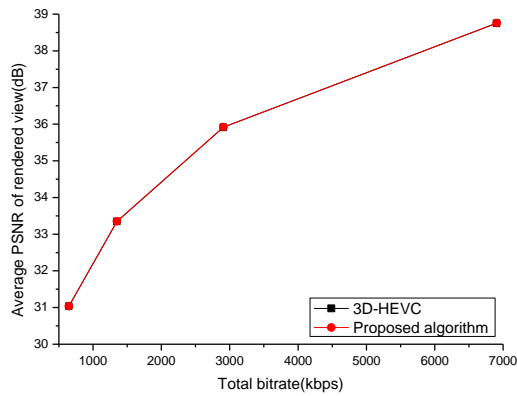
Sequences	Texture video		Rendered view		Dtime (%)
	BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)	
Kendo	0.68	-0.02	0.22	-0.01	-29.4
Balloons	0.64	-0.02	0.19	-0.01	-28.9
Newspaper	0.89	-0.03	0.39	-0.02	-30.8
Shark	1.23	-0.04	0.58	-0.02	-37.2
Undo_Dancer	1.16	-0.04	0.52	-0.02	-36.5
GT_Fly	0.71	-0.02	0.26	-0.01	-31.2
Poznan_Street	0.27	-0.01	0.11	-0.01	-29.8
Poznan_Hall2	0.18	-0.01	0.04	-0.00	-23.7
Average	0.72	-0.02	0.29	-0.01	-30.9



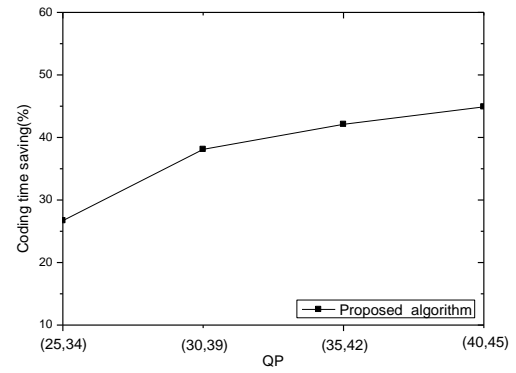
(a) RD curve of "Newspaper"



(b) Time saving curves of "Newspaper"



(c) RD curve of "Shark"



(d) Time-saving curves of "Shark"

Figure 2. Experimental results of "Newspaper" and "Shark"

For evaluating the performance in detail, the results of our algorithm compared to some fast 3D-HEVC methods is shown in Table 8. There are comparison results of the state-of-the-art fast method online learning-based complexity reduction scheme (OLCRS) [11] for 3D-HEVC. The proposed algorithm outperforms it, with an average of more than 14.1% saved time. The computational reduction is very large because the properties of the texture motion homogeneity are fully applied to select the prediction mode, while some candidate prediction modes are not tested in the 3D-HEVC system. Meanwhile, the average loss of RD performance is negligible. Therefore, our algorithm significantly reduces computational burden while maintaining similar RD performances as the initial 3D-HEVC.

Table 8. Coding results of the proposed algorithm compared to the OLCRS [11] method

Sequences	Texture video		Rendered view		Dtime (%)
	BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)	
Kendo	0.45	-0.02	0.12	-0.01	-11.5
Balloons	0.52	-0.02	0.09	-0.01	-10.1
Newspaper	1.01	-0.03	0.39	-0.02	-11.2
Shark	1.91	-0.05	0.56	-0.02	-19.7
Undo_Dancer	1.84	-0.05	0.95	-0.03	-18.5
GT_Fly	0.92	-0.03	0.11	-0.01	-11.6
Poznan_Street	0.21	-0.01	0.04	-0.00	-12.2
Poznan_Hall2	0.12	-0.01	0.02	-0.00	-17.8
Average	0.87	-0.03	0.29	-0.01	-14.1

## 5. Conclusions

In the paper, we propose a fast depth level range determination algorithm based on motion homogeneity to alleviate the computational burden of the 3D-HEVC system. Our algorithm is performed on the recent 3D-HEVC reference software. Experimental results show that our algorithm can reduce encoding time by approximately 30.9% on average compared to the HTM 16.0 system, while limiting the loss of RD performance. Furthermore, it outperforms the recent fast 3D-HEVC methods with an additional saving of encoding time.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 61771432 and 61302118), Scientific Project (No. 182102210156 and 182102210610), Innovation Talents (No. 17HASTIT022), Young Key Teacher (No. 2016GGJS-087), and Education Department Project (No. 18B510019 and 17B510011).

## References

1. A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," in *Proceedings of the IEEE*, Vol. 99, No. 4, pp. 626-642, April 2011
2. G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and System for Video Technology*, Vol. 22, No. 12, pp. 1649-1668, December 2017
3. G. Tech, Y. Chen, K. Müller, J. Ohm, and A. Vetro, "Overview of the Multiview and 3D Extensions of High Efficiency Video Coding," *IEEE Transactions on Circuits and System for Video Technology*, Vol. 26, No. 1, pp. 35-49, January 2016
4. Y. Chen, G. Tech, K. Wegner, and S. Yea, "Test Model 11 of 3D-HEVC and MV-HEVC," Joint Collaborative Team on 3D Video Coding Extensions (JCT-3V) document *JCT3V-K1003*, 11th Meeting: Geneva, CH, 12-18 February 2015
5. K. Müller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D High Efficiency Video Coding for Multi-View Video and Depth Data," *IEEE Transactions on Circuits and System for Video Technology*, Vol. 22, No. 9, pp. 3366-3378, September 2013
6. L. Ding, P. Tsung, S. Chien, W. Chen, and L. Chen, "Content-Aware Prediction Algorithm with Inter-View Mode Decision for Multiview Video Coding," *IEEE Transactions on Multimedia*, Vol. 10, No. 8, pp. 1553-1563, December 2008
7. W. Zhu, X. Tian, F. Zhou, and Y. Chen, "Fast Inter Mode Decision based on Textural Segmentation and Correlations for Multiview Video Coding," *IEEE Transactions on Consumer Electronics*, Vol. 56, No. 3, pp. 1696-1704, August 2010
8. Y. Zhang, S. Kwong, G. Jiang, X. Wang, and M. Yu, "Statistical Early Termination Model for Fast Mode Decision and Reference Frame Selection in Multiview Video Coding," *IEEE Transactions on Broadcasting*, Vol. 58, No. 1, pp. 10-23, March 2012
9. T. Zhao, S. Kwong, H. Wang, Z. Wang, Z. Pan, and C. C. J. Kuo, "Multiview Coding Mode Decision with Hybrid Optimal Stopping Model," *IEEE Transactions on Image Processing*, Vol. 22, No. 4, pp. 1598-1609, April 2013
10. L. Zhang, Y. Chen, X. Li, and S. Xue, "Low-Complexity Advanced Residual Prediction Design in 3D-HEVC," in *Proceedings of 2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp.13-16, June 2014
11. H. R. Tohidypour, M. T. Pourazad, and P. Nasiopoulos, "Online-Learning-based Complexity Reduction Scheme for 3D-HEVC," *IEEE Transactions on Circuits and System for Video Technology*, Vol. 26, No. 10, pp. 1870-1883, October 2016
12. Q. Zhang, H. Chang, Q. Wu, and Y. Gan, "Fast Motion and Disparity Estimation for HEVC based 3D Video Coding," *Multidimensional Systems & Signal Processing*, Vol. 27, No. 3, pp. 743-761, July 2016